

What is claimed is:

1. A method for accessing resources within a data processing network, comprising the steps of:

5        computing a set of hash values representing a set of resources stored in association with  
at least one data processing system within the network;  
         storing the computed set of hash values;  
         in response to a requirement for access to a first resource which is accessible via a  
bandwidth-sensitive connection, retrieving a hash value derived from the required first resource;  
10        comparing the retrieved hash value with the stored set of hash values to identify a match  
between the retrieved hash value and any of the stored set of hash values;  
         in response to identifying a match for the retrieved hash value, initiating retrieval of the  
required first resource from said at least one data processing system; and, if no match is  
identified for the retrieved hash value, retrieving the required first resource via said bandwidth-  
15        sensitive connection.

2. The method of claim 1, wherein the step of retrieving the hash value derived from the  
required first resource comprises:

         sending a resource access request to a server computer via the bandwidth-sensitive  
20        connection; and  
         receiving the hash value from the server computer via the bandwidth-sensitive  
connection.

3. The method of claim 1, further comprising the step of:

25        initiating retrieval of the required resource via said bandwidth-sensitive connection in  
parallel with initiating retrieval of the required first resource from said at least one data  
processing system.

4. The method of claim 3, further comprising the step of:

responding to receipt of said required first resource from either one of said bandwidth-sensitive connection or said at least one data processing system by terminating in-progress retrieval operations relating to said required first resource.

5     5.     The method of claim 3, wherein the step of initiating retrieval of the required resource from said at least one data processing system comprises initiating retrieval of resource size information and initiating retrieval of the bit sequence of said resource in a reverse order relative to the retrieval of said resource via the bandwidth-sensitive connection; and combining portions of the bit sequence of said resource received via the bandwidth-sensitive connection and received  
10    from said at least one data processing system to build the bit sequence of said resource.

6.     A method for accessing resources within a data processing network, comprising the steps of:

15         computing a set of hash values representing a set of resources distributed across a plurality of data processing systems within a local area network (LAN), the resources within said set of resources being accessible from respective ones of the plurality of data processing systems;

       storing the set of hash values together with an identification of a respective data processing system of said plurality of data processing systems storing the resource corresponding to each of the set of hash values;

20         in response to a requirement for access to a resource which is stored at a remote data processing system, retrieving from the remote data processing system a hash value derived from the required resource;

       comparing the retrieved hash value with the stored set of hash values to identify a match between the retrieved hash value and any of the stored set of hash values;

25         in response to identifying a match for the retrieved hash value, initiating retrieval of the required resource from a respective one of the plurality of data processing systems at which the resource corresponding to the matched hash value is stored; and, if no match is identified for the retrieved hash value, retrieving the required resource from said remote data processing system.

7. The method of claim 6, wherein the set of hash values and identification of a respective data processing system are stored with information regarding the location within storage of the respective data processing system of the resource corresponding to the hash value.

5 8. A method for accessing resources within a data processing network, comprising the steps of:

computing a set of hash values representing a set of resources distributed across a plurality of data processing systems within the network, the resources within said set of resources being accessible from respective ones of the plurality of data processing systems;

10 storing the set of hash values together with an identification of a respective data processing system of said plurality of data processing systems storing the resource corresponding to each of the set of hash values;

in response to a requirement for access to a resource which is accessible via a bandwidth-sensitive connection, retrieving a hash value derived from the required resource;

15 comparing the retrieved hash value with the stored set of hash values to identify a match between the retrieved hash value and any of the stored set of hash values;

in response to identifying a match for the retrieved hash value, retrieving the required resource from a respective one of the plurality of data processing systems at which the resource corresponding to the matched hash value is stored; and, if no match is identified for the retrieved hash value, retrieving the required resource via the bandwidth-sensitive connection.

9. A data processing apparatus comprising:

a data processing unit;

25 a data storage unit for storing a set of hash values representing a set of resources available from a first set of data processing systems; and

a repository manager for comparing a received hash value with the stored set of hash values to identify a match between a received hash value and any of the stored hash values,

wherein the repository manager is configured to respond to identification of a match between a stored hash value and a received hash value representing a required resource, by

30 outputting a confirmation that the required resource is available from a data processing system of the first set of data processing systems.

10. The data processing apparatus of claim 9, wherein the repository manager is configured to respond to identification of a match between a stored hash value and a received hash value representing a required resource, by outputting an identification of the location of the required resource at a data processing system of the first set of data processing systems.

11. The data processing apparatus of claim 9, wherein the repository manager is configured to perform said comparison in response to a request for a resource from a requestor program, and wherein the repository manager is configured to respond to identification of a match between a stored hash value and a received hash value representing a required resource by obtaining a copy of the required resource and forwarding the obtained copy to the requestor program.

12. The data processing apparatus of claim 9, further comprising:  
a secure hashing function for computing hash values representing resources that are available from the first set of data processing systems.

13. A data processing apparatus comprising:  
a data processing unit;  
a secure hashing function for computing a hash value representing at least one resource;  
a data storage unit for storing said at least one resource in association with the computed hash value; and  
a resource access manager configured to respond to a request from a requestor for access to said at least one resource by sending to the requestor the computed hash value representing the at least one resource.

14. A data processing apparatus comprising:  
a data processing unit;  
a data storage unit; and  
a resource access requestor configured to respond to a requirement for a resource which is accessible via a bandwidth-sensitive connection, by retrieving a secure hash value representing the required resource and sending the hash value representing the required resource to a

repository manager, to initiate comparison between the retrieved hash value and stored hash values representing resources accessible on a first set of data processing systems and thereby to determine whether the retrieved hash value matches any of the stored hash values, wherein the resource access requestor is configured to respond to identification of a match between the  
5 retrieved hash value and a stored hash value by initiating retrieval of the desired resource from a data processing system of the first set of data processing systems.

15. A data processing network comprising a plurality of pool server data processing systems, each comprising:

10 a data processing unit and a data storage unit for storing a set of hash values representing a set of resources available via the respective pool server data processing system, and a repository manager configured to compare a received hash value with the stored set of hash values to identify a match between a received hash value and any of the stored hash values, wherein the repository manager is configured to respond to identification of a match between a  
15 stored hash value and a received hash value representing a required resource by outputting a confirmation that the required resource is available via the respective pool server data processing system.

16. The data processing network of claim 14, further comprising:

20 a set of client data processing systems each comprising a data storage unit storing at least one resource, wherein each client data processing system is associated with a pool server data processing system and the repository manager of each pool server data processing system is configured to store hash values corresponding to a set of resources distributed across the respective set of associated client data processing systems.

25 17. A computer program product comprising program code recorded on a recording medium, the program code comprises a repository manager for comparing a received hash value with the stored set of hash values to identify a match between a received hash value and any of the stored hash values, wherein the repository manager is configured to respond to identification of a match  
30 between a stored hash value and a received hash value representing a required resource, by

outputting a confirmation that the required resource is available from a data processing system of the first set of data processing systems.

18. A computer program product comprising program code recorded on a recording medium,  
5 the program code comprises a resource access requestor configured to respond to a requirement for a resource which is accessible via a bandwidth-sensitive connection, by retrieving a secure hash value representing the required resource and sending the hash value representing the required resource to a repository manager to initiate comparison between the retrieved hash value and stored hash values representing resources accessible on a first set of data processing  
10 systems, thereby to determine whether the retrieved hash value matches any of the stored hash values, wherein the resource access requestor is configured to respond to identification of a match between the retrieved hash value and a stored hash value by initiating retrieval of the desired resource from a data processing system of the first set of data processing systems.
- 15 19. A method for accessing resources within a data processing network, comprising the steps of:  
computing a hash value representing a resource stored at a server data processing system;  
and  
embedding the hash value within a hyperlink of a data entity associated with the resource.
- 20 20. The method of claim 19, further comprising:  
embedding at least one additional resource reference within the hyperlink, to generate a hyperlink that comprises a plurality of resource references.
21. The method of claim 20, further comprising:  
25 embedding within the hyperlink relative priorities associated with each of the plurality of resource references.
22. The method of claim 19, wherein the steps of computing a hash value and embedding the hash value in a hyperlink are performed in response to a request for access to the data entity.

23. The method of claim 19, wherein the steps of computing a hash value and embedding the hash value in a hyperlink are performed by a preprocessor which scans data entities for hyperlinks, determines the hash value of a resource referenced by the hyperlink, and embeds the determined hash value within the hyperlink.

5

24. A method for accessing resources within a data processing network, comprising the steps of:

storing a set of hash values computed from a set of resources, wherein the resources are available from at least one data processing system within the network;

10 receiving a hash value representing a desired resource, and comparing the received hash value with the stored set of hash values to identify a match between the received hash value and any of the stored hash values; and

in response to identifying a match for the received hash value, initiating retrieval of the required resource from said at least one data processing system.

15

25. The method of claim 24, further comprising:

in response to failure to identify a match for the received hash value by comparison with the set of hash values, initiating a communication with a data processing apparatus storing a second set of hash values and initiating a comparison between the received hash value and the

20 hash values of the second set of hash values.

26. A method for accessing resources within a data processing network, comprising the steps of:

25 receiving a data entity including a hyperlink that references a server computer and references a resource by means of a hash value representing the resource;

in response to user selection of the hyperlink, sending a resource-access request including the hash value to the referenced server computer.

27. The method of claim 26, further comprising the steps of:

30 for a received data entity including a hyperlink that comprises a plurality of resource references and associated relative priorities of each of the plurality of resource references,

determining the resource reference having the highest priority, identifying a server computer from the highest priority resource reference, and sending to the identified server computer a resource-access request including the highest priority resource reference.

5 28. The method of claim 27, further comprising the steps of:

in response to a notification that the resource identified by the highest priority resource reference is inaccessible via the server computer, determining the resource reference having the second highest priority, identifying a respective server computer from the second highest resource reference, and sending to the respective identified server computer a resource-access  
10 request including the second highest resource reference.

29. A method for computing an identifier for a resource, comprising the steps of:

scanning the resource to identify hash-based URIs within hyperlinks embedded in the resource;

15 extracting the hash-based URIs from the hyperlinks to generate a modified resource which does not contain the extracted hashes; and

applying a secure hash function to the modified resource to compute a hash value representing the modified resource.

20 30. A method for transferring directory-structured data from a first data processing system to a second data processing system, comprising the steps of:

computing a set of hash values representing a set of resources stored in a directory at the first data processing system;

25 generating a directory template representing the structure of the directory including the locations within the directory of the resources represented by the set of hash values;

sending the set of hash values and the directory template to the second data processing system;

30 comparing the set of hash values received at the second data processing system with stored hash values representing resources accessible from a system other than the second data processing system to identify a match between a received hash value and a stored hash value; and



in response to identifying a match between a received hash value and a stored hash value, accessing from said other system the resource represented by the matching hash value.

31. A method for restoring computer system resources following a failure, comprising the steps of:

for each of a set of data processing systems, computing a set of hash values representing a set of resources stored within the respective data processing system;

storing the set of hash values for each of the set of data processing systems in at least one repository;

in response to a failure affecting resources stored at a data processing system within the set of data processing systems, comparing the set of hash values stored in the repository for the affected data processing system with the stored hash values for other data processing systems within the set of data processing systems, to identify resources stored within said other data processing systems having hash values matching resources of the affected data processing system; and

restoring, to the affected data processing system from said other data processing systems, identified resources having hash values matching resources of the affected data processing system.

32. A method for restoring computer system resources, comprising the steps of:

for each of a set of data processing systems, computing a set of hash values representing a set of resources stored within the respective data processing system;

storing the set of hash values for each of the set of data processing systems in at least one repository, together with a timestamp relating to the time at which the hash value was computed and a path within the respective data processing system at which the resource represented by the hash value is located;

in response to a requirement for a particular version of a resource at a particular time, using the hash value, timestamp and path stored in the repository to identify a relevant resource.